UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

DTIC FILE COPY

(1)

AD-A196 688

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFIT/CI/NR 88- 6	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) PRIORITIZING REQUIREMENTS FOR COMMAND AND CONTROL SYSTEMS		5. TYPE OF REPORT & PERIOD COVERED MS THESIS
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) PHILLIP DREW PROSSEDA		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS AFIT STUDENT AT: HARVARD UNIVERSITY		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS		12. REPORT DATE 1988
		13. NUMBER OF PAGES 46
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) AFIT/NR Wright-Patterson AFB OH 45433-6583		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

DISTRIBUTED UNLIMITED: APPROVED FOR PUBLIC RELEASE

DTIC ELECTE AUG 0 3 1988 S D H

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

SAME AS REPORT

18. SUPPLEMENTARY NOTES

Approved for Public Release: IAW AFR 190-1
LYNN E. WOLAVER
Dean for Research and Professional Development
Air Force Institute of Technology
Wright-Patterson AFB OH 45433-6583

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

ATTACHED

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

MAJOR REPORT ABSTRACT

Author: Phillip Drew Prosseda, Captain, USAF

Title: Prioritizing Requirements for Command and Control Systems

Date: 1988

Length: 46 pages

Degree: Master in Public Policy

Institution: John F. Kennedy School of Government, Harvard University

This report focuses on Electronic System Division's role in prioritizing command and control system requirements during the concept formulation phase. The first section of the report discusses institutional constraints that impede requirement prioritization. The roles of the developer, user, and support agencies are examined and three possible approaches to harmonize conflicting interests are presented. The second section of the report deals more directly with technical impediments to requirement p[illegible]ritization. The evolutionary nature of command and contr[illegible] ystems interacting with traditional acquisition a[illegible] [illegible]aches is presented as a major stumbling block to requirement prioritization. This section concludes with a fresh look at prototyping as a possible solution to the problem.

PRIORITIZING REQUIREMENTS

for

COMMAND AND CONTROL SYSTEMS

prepared for:

Colonel Robert W. Drewes
Deputy Commander for Contracting
Electronic Systems Division

by:

Phillip Drew Prosseda

Harvard University
John F. Kennedy School of Government
April 12, 1988

TABLE OF CONTENTS

EXECUTIVE SUMMARY

This report focuses on Electronic System Division's (ESD) role in prioritizing command and control (C2) system requirements during the concept formulation phase. Particular attention is paid to this phase of development because it is during this phase that alternative system concepts are tested against user requirements and key trade-offs are made between cost and performance. Decisions made at this point have an enormous impact on user satisfaction and the ultimate cost of C2 systems. The developer, Electronic Systems Division, must thoroughly understand the system requirements and priorities in order to successfully complete conceptual design.

THE PROBLEMS AND FINDINGS

All too often, ESD finds that it does not adequately understand the system requirements and priorities of C2 systems. This happens because the developer is performing a number of functions at the same time and is attempting to coordinate requirements with many different organizations. In some instances, the developer's functions are both complimentary and contradictory. To more clearly understand the developers's tasks in prioritizing system requirements, a distinction is made between the developer's role as purchasing agent and the developer's role as supplier.

As a purchasing agent, the developer is attempting to acquire the best value by balancing requirements to minimize total expenditures. This function involves many participants, all with specialized knowledge and conflicting interests. The user, who is the prime originator of requirements, is concerned with performance. Other participants in the requirement process are mainly concerned with support requirements. Given a limited budget, some requirements conflict. Many of the participants in the requirement process find it easier not to prioritize requirements in this environment. They prefer to let the developer decide these issues. However, the developer does not have complete authority to decide. Any judgement he makes is subject to review by the participants and can be reversed or delayed if a participant fights strongly enough.

As a supplier, the developer is confronted with a set of different issues. In this case, the unique nature of command and control systems and the traditional acquisition structure make it difficult for the user to understand and communicate his requirements. This, in turn, results in ill defined requirements that are almost impossible to prioritize. Because of the sophisticated nature of command and control technology, the practice of using user surrogates to specify requirements insulates the real user from the technology that he must understand. In practice, the only effective way to define C2 requirements is to allow the real user to react to the technology and iterate his requirements through "hands on experience".

Priorities are best determined through an iterative process that involves the real user, user surrogate, and the developer in an integrated rather than sequential manner.

RECOMMENDATIONS

Recommendation 1: Electronic Systems Division should aggressively use design to cost and design to life cycle cost techniques to help prioritize requirements.

ESD has the ability to determine rough fiscal constraints by using costs of similar systems and assessing the political environment to determine funding limits. Once a rough constraint is established, ESD can employ design to cost techniques to determine cost drivers for the proposed system. The cost drivers then provide the ability to challenge the participants' requirements and determine their utility. The knowledge of cost combined with performance and support considerations can then be used to prioritize requirements.

Recommendation 2: Electronic Systems Division should not assume complete authority for determining requirement priorities at this time.

Much has been said about fixing responsibility in the acquisition process. In the case of requirement prioritization, ESD seems to have the broadest view of all the participants' interests and is in the most logical position to assume responsibility for these decisions. However, assuming more of this responsibility cannot be accomplished unilaterally. Many vested interests and legitimate concerns of the other participants in the requirement process will have to be addressed before this can be accomplished.

Recommendation 3: Electronic Systems Division should develop a rapid prototyping capability to define and prioritize user requirements.

It is apparent that the nature of C2 systems along with traditional acquisition methods have prevented adequate expression of requirements. ESD can help solve this problem by using prototypes installed at the real user's location to determine requirements and priorities. This approach provides a better match between the developer's technical expertise and the real user's operational expertise. Prototypes allow requirements to be iterated and the developer can gain a better understanding of the real user's true priorities.

INTRODUCTION

The concept formulation phase is the most critical part of the acquisition process. It is during this stage of development that alternative concepts are tested against user requirements and key trade-offs are made between cost and performance. Up to 85% of the projected life cycle cost of a weapon system results from the decisions made during conceptual design [1]. If the concept formulation phase is not completed properly, subsequent phases such as demonstration and test will reveal deficiencies in requirements and technical approaches that will require substantial changes and result in delays to the program.

Successful conceptual design rests on a set of well defined and prioritized system requirements. These requirements originate with the user and must be fully understood by the developer. If the developer does not understand the user's requirements and his priorities, it will be nearly impossible to translate the requirements into meaningful contractual language in the request for proposal. Contractors rely on this translation to formulate alternative system designs. Figure 1 depicts the preliminary steps of the concept formulation phase and illustrates how concept definition depends on operational requirements.

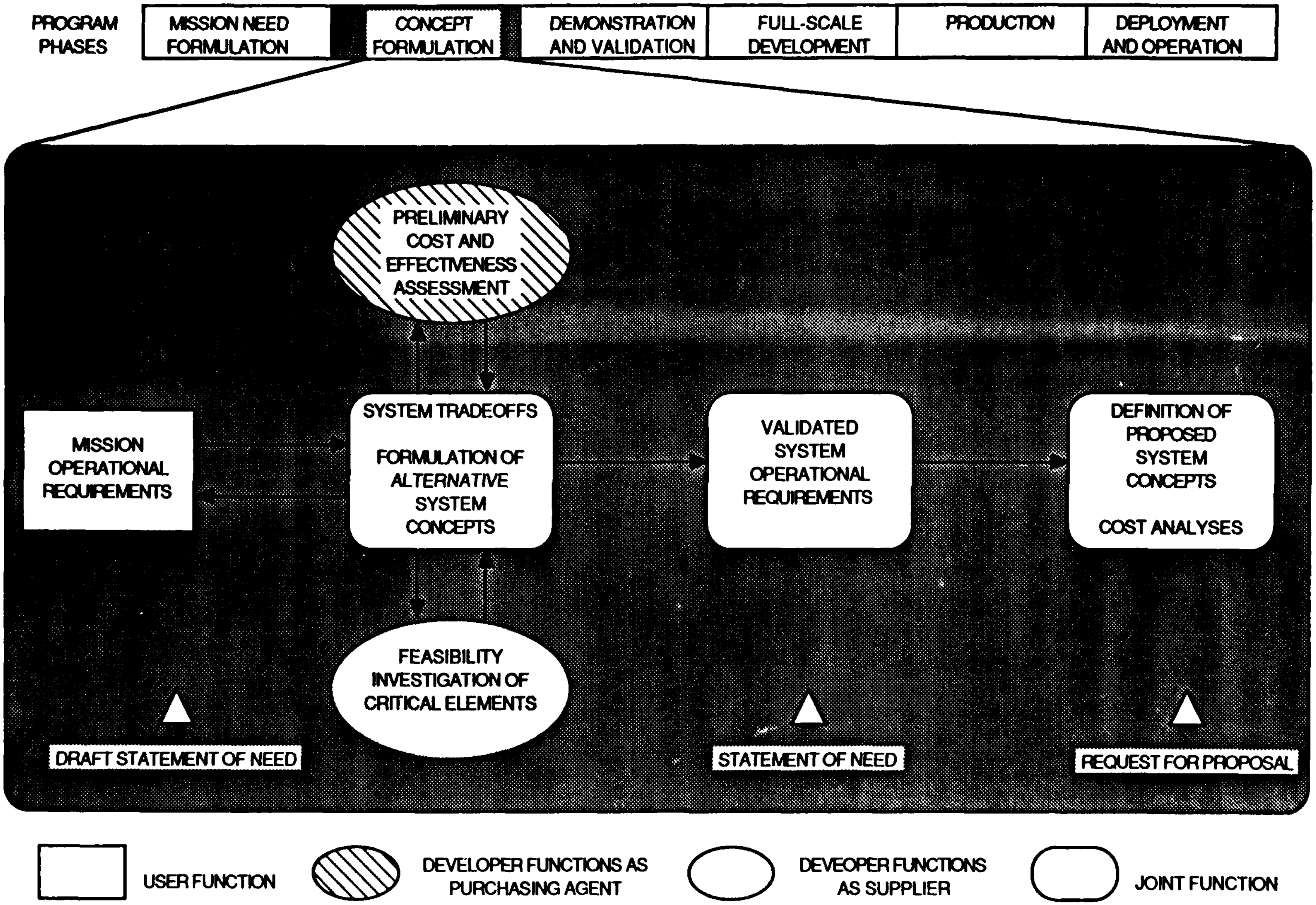

FIGURE 1. REQUIREMENT VALIDATION AND THE CONCEPT FORMULATION PHASE [2]

As can be seen in the diagram, the process labeled system trade-offs and formulation of alternative system concepts depends on two functions performed by the developer. First, the developer must perform cost and effectiveness assessments to make overall trade-offs on system requirements. Second and simultaneously, the developer must perform a feasibility investigation of the critical elements (mostly performance characteristics) of the program. The first function resembles that of the traditional purchasing agent. In this case the developer is attempting to acquire the best value by balancing requirements to minimize total expenditures. The second function is more similar to the role of a supplier in the commercial world. The developer is providing specific technical advice to help the user solve his problem. While these two functions are qualitatively different, they occur simultaneously and both require prioritized requirements for successful completion.

Both the literature and interviews indicate that there is a clear lack of requirement prioritization in systems acquisition [3]. Electronic Systems Division personnel have also revealed that lack of prioritization results in production of command and control (C2) systems that do not meet the user's needs. As shown above, prioritized requirements are necessary to direct the research and development community during design studies and they assist in allocating scarce funds to the most important uses in a system. Given the ultimate impact on a system's military capability and cost, it is essential that requirements be prioritized. Without a set of prioritized requirements, the developer and contractor cannot assess alternative designs and produce a system that meets the user's needs.

This paper will analyze impediments to requirement prioritization and recommend solutions that Electronic Systems Division (ESD) can employ to produce prioritized operational requirements. The first section of the paper concentrates on requirement prioritization and the developer's role as purchasing agent. In this case, the developer must examine both operational performance requirements and operational suitability requirements, weigh them with respect to cost and performance, and produce a ranking that maximizes overall utility of the system. Here, the impediments to requirement prioritization are mainly institutional in nature. The second section addresses requirement prioritization and the developer's role as supplier. In this situation, the developer is attempting to help the user define and prioritize his performance requirements so these requirements can be viewed in the larger framework discussed in the first section. The developer's role as supplier is complicated by the unique nature of command and control systems. Furthermore, institutional arrangements also contribute to the problem of requirement prioritization. Approaches to the problem and recommendations are presented at the end of each section. Finally, the conclusion draws both problems together and puts them in perspective.

THE DEVELOPER AS PURCHASING AGENT

A. ANALYSIS OF THE PROBLEM

1. Documenting and Coordinating Operational Requirements

The requirement for a new Air Force weapon system starts when the user identifies an operational deficiency that cannot be corrected "through changes in tactics, strategy, doctrine, or training and whose solution requires a new development or upgrade of an existing system" [4]. System requirements are formally documented in a Statement of Operational Need (SON) which is circulated in a draft version to numerous Air Force agencies for comment.

Circulation of the draft SON is necessary to obtain a broad Air Force perspective on a proposed system. Conceptually, this is a sound process. Inputs are not only needed from the people who will operate the system, but also from the various agencies that will deal with a new system during its life cycle. Many requirements must be coordinated. For example, testing criteria must be established, maintenance concepts formulated, logistic support planned, and manpower training requirements evaluated. As suggested, this is a huge planning task and in many instances is performed at a grass roots level.

In total, 18 Air Force agencies or commands are listed as action addressees (i.e. These organizations must make formal comments on any draft SON) and 53 organizations receive an information copy of the draft SON (See Attachment 1). These agencies are responsible for reviewing the requirements listed in the SON and determining if the proposed system will affect their operations. If a new system does affect an agency, it must comment on what the effect will be and state any constraints or additional requirements.

2. Participants and Their Interests

In practice, circulating a draft SON helps prevent errors of omission but it does not prevent errors of commission. Because many of the SON reviewers are grass roots workers, they tend to have a highly parochial view and do not understand how including some requirements affects the entire system. Headquarters Air Force reviews the draft SON and attempts to reconcile conflicting requirements, but this process occurs before any significant information on cost is obtained. Therefore, many draft SON's still contain a multitude of requirements promulgated by many different agencies.

Getting all these agencies to agree on a prioritized list of requirements can be extremely difficult. Not only are many of the agencies geographically disparate, they also have differing functions and responsibilities. The user is mainly concerned with the system's performance and how the system's performance helps him accomplish his wartime mission. The support agencies are concerned with an entity called operational suitability. The operational suitability of a system determines the level of effort and amount of resources that support agencies must devote to maintaining a system.

At this point, it is necessary to define the military worth of a system and explain the factors that influence this measure of value. Military worth is the overall utility of the system, that is, the degree to which the system performs the intended mission in both peacetime and war. Military worth depends on the operational performance and the operational suitability characteristics of the system.

MILITARY WORTH = OPERATIONAL PERFORMANCE + OPERATIONAL SUITABILITY

Performance is a relatively straight forward concept. However, operational suitability is a catchall term that includes everything else that may matter during a system's lifetime. Department of Defense Directive 5000.3 defines operational suitability as, "The degree to which a system can be placed satisfactorily in field use, with consideration being given to availability, compatibility, transportability, interoperability, reliability, wartime usage rates, maintainability, safety, human factors, manpower supportability, logistic supportability, and training requirements" [5]. While performance is the exclusive domain of the user, most of the "ilities" come under the purview of separate organizations created to deal with subsets of operational suitability.

Conventional wisdom dictates that the degree to which operational performance and operational suitability are factored into the military worth equation depends on the type of system and how it is employed. However, the resulting balance can also be viewed as the outcome of a struggle between the user and organizations responsible for operational suitability. Description of two of the participants in the requirement process will help illuminate the tension that exists.

The user: The user is generally a major command (MAJCOM) in the Air Force with an operational mission that requires C2 systems to control its forces. Users include Strategic Air Command, Tactical Air Command, Military Airlift Command and the North American Defense Command.

The user is concerned with performance of a new system and how increased performance will help him accomplish the mission. For this reason and also because the user is the prime originator of the SON, requirements for operational performance are emphasized over less understood operational suitability requirements. The user can evaluate the military worth of performance requirements and only vaguely comprehends the utility of operational suitability requirements. The user is aware of operational suitability to the extent that factors such as reliability and maintainability affect the performance of the system. However, factors such as training maintenance technicians and supplying spare parts are secondary concerns in the initial stages of acquisition. The user wants a system that will meet the current and projected threat. In addition, because the user is the one who employs the system in combat, his opinion generally receives more attention than that of support organizations.

Air Force Logistics Command: Air Force Logistics Command is the organization that is responsible for maintaining systems and correcting design deficiencies once the initial acquisition is complete. To a large extent, the organizational culture of Logistics Command has been influenced by the systems they have maintained. One trend that has become more pronounced with the passage of time is the increased service life of weapon systems in the Air Force inventory [6]. Many of these systems have been extended well past their planned service life and do not match the potential reliability available with new technology. Experience with less reliable systems that have been maintained well past their expected service life has made Air Force Logistics Command an advocate for improved operational suitability. Logistics Command bears most of the cost of poorly designed systems. Retrofit of systems that do not meet user requirements, that are unreliable, or that have poor maintainability characteristics, not only imposes financial costs but also diverts scarce talent from other projects.

3. Conflicting Interests

Although it is in the long term interest of the user to acquire systems that are more reliable and maintainable, this often does not occur because of the nature of defense funding. Congress votes separate funds for acquisition and for operations and maintenance. Acquisition appropriations are highly visible, whereas operations and maintenance funds are costs that must be paid for equipment already in the inventory. Because the user is concerned with the performance necessary to meet a projected threat, he is hesitant to include specific reliability and maintainability requirements for fear that it will drive up the acquisition cost of the program. If these requirements significantly raise costs, a program could be cancelled and the user would receive no new capability to meet the threat [7]. Logistics Command bears most of the maintenance costs of less reliable systems. Therefore, it fights for tougher reliability and maintainability requirements. If the conflicting requirements are prioritized, then one or both of the parties interests will be compromised. Although recent high level policy guidance has attempted to internalize the costs of less reliable and maintainable systems, this example is cited to illustrate why many of the costs of systems acquisition are not internalized. When many actors enter the requirements process, and all lay claim to urgent priorities, it is difficult to reach an agreement on a ordered list of requirements.

In the early stages of the requirement process the user is the lead organization. The user documents his requirements in the Statement of Need and coordinates them with the various supporting organizations. At this point, the developer's role is mainly to provide technical assistance to identify solutions to the user's problem. While cost is a consideration, it is not predominant. This is because the user's currency of evaluation is performance and trade-offs are more likely to be viewed from this perspective. As the requirements process continues, the developer begins to takeover as the lead organization. Stated requirements are now viewed from the perspective of the developer and cost becomes the currency used to evaluate trade-offs. The developer, however, never gains complete control of the process. He must always proceed with the concurrence of the user.

When cost becomes a major factor in trade-off decisions, the user is forced to employ strategies that will help maintain the performance orientation of the system's development. If the user adamantly objects to a trade-off based on cost, he can veto the developer's decision. This approach is useful only as a last resort. If it is over employed, the developer's support for the program wanes and relationships between the user and developer are stressed. A less threatening strategy is non-prioritization of requirements. This strategy is a way out of the trade-off dilemma because it offers the hope that increased funding can be approved and trade-off decisions based on cost can be avoided.

Non-prioritization of requirements is a strategy available to the user (as well as other organizations in the requirements process) because of the dual role of the developer. On the one hand, the developer is responsible for providing the engine ring and programmatic expertise necessary to acquire weapon systems from contractors. In this role the developer is charged with meeting the user's (and other organization's) requirements. On the other hand, the developer must price the system requirements, recommend funding levels, and then work within funding levels approved by Headquarters Air Force and Congress. If the developer completely understands the user's requirements and priorities, the user is more vulnerable to funding instability. As the budget is cut, the developer may be less resistant to cuts in a program that he knows can be managed with less resources (That is, a program in which the developer will have less difficulty finding the features to cut.).

This situation occurs whenever approved funding is less than that necessary to meet validated requirements. The user has difficulty appreciating the funding constraint because his perspective is that he has requirements and the requirements must be met to deter or combat the threat. Users feel that prioritization of requirements would allow program managers to delete the lowest priorities from a system every time the budget is cut or technical difficulties require more unavailable resources. From the user's perspective, requirements are still requirements regardless of their ranking on a list.

Figure 2 is included to help illustrate the major participants and their relationships in both the requirement validation and funding process for a new weapon system. As shown, the user has primary knowledge of the mission, operating environment, and operational performance requirements and is initially charged with developing the Statement of Need. The various supporting commands, indicated in boxes, understand the operational suitability requirements and other support requirements affecting the acquisition. The developer has the programmatic expertise to perform cost and technical assessments of the system requirements and the authority to procure the system from contractors. Headquarters Air Force reviews the requirements and uses cost estimates established by the developer to plan funding. Headquarters also determines the relative priority of a system compared to other systems competing for funding and submits the system for funding competition. Finally, Congress determines ultimate program funding through the appropriation process.

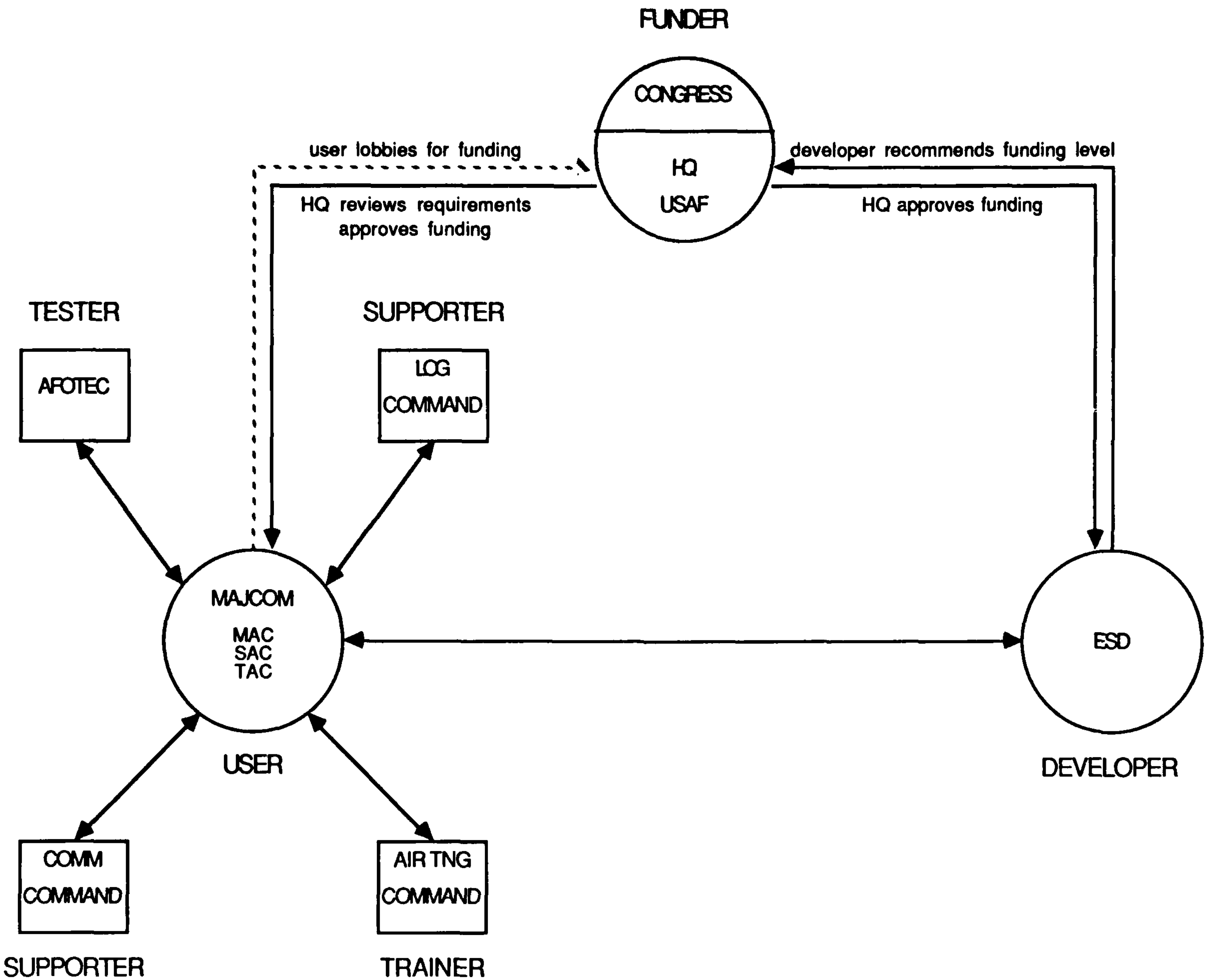

FIGURE 2 . MAJOR ACTORS IN THE REQUIREMENTS PROCESS

In this depiction, the developer is in an awkward position. He must recommend funding levels to the Air Staff, but before he can accomplish this task he must understand the system requirements and priorities. Priorities are difficult to establish because the user and various support commands recognize that the developer is the funding advocate for the system. The user has leverage over the developer because in the scheme of things the developer must be responsive to the user's needs. Since the user does not have direct control over funding he does not have the motivation to make requirement trade-offs based on a funding constraint. In many instances, the user does not even have the capability to assess the trade-offs in this manner.

B. APPROACHES TO THE PROBLEM

1. Provide Better Cost Information

It is the Packard Commission's view that the process described in the preceding section leads to overstated military requirements. These requirements are translated into specifications and the result is a system that costs far more than it should. The commission also feels that cost is not considered early enough in the acquisition process. The commission states:

> In general, this process does not adequately involve participants with a sophisticated knowledge of the cost and schedule implications of technical improvements required to satisfy these characteristics. ... If users understood the likely impact of their requirements on the schedule, quantity, and maintainability of the weapons they eventually received, they would have a strong motivation for compromise. Generally, however, that compromise--a conscious trade-off between performance and cost--does not take place to an adequate degree. Implicitly, it is assumed that military requirements should be "pure" and that any necessary trade-offs will take place later in the process [8].

Given this view, the developer must find a way to use cost as a method for evaluating requirement priorities. Once this information is known, users and support organizations may be able to prioritize requirements and make the necessary trade-offs early in the concept formulation phase.

This problem has long been recognized within the Department of Defense and a large amount of literature has been published on the need to design systems to a proposed budget or cost. Design to cost is a concept borrowed from commercial product development. In the commercial world cost targets are a function of market demand. The commercial firm estimates what consumers are willing to pay for a new product with certain features and then uses this market analysis to set a cost goal for the unit price of the product. This provides the proper incentive for the designers to make appropriate design trade-offs to keep the cost of the product on target. In the defense world this process is reversed. First, the requirements for a new system are established and then a cost for the system is established. Since the user provides the market for a system, there is no market risk, and the need to produce at a certain unit price is not as urgent. In addition, since the user does not control funding for his system he does not have the same cost incentive to perform trade-offs. In general, the user lacks the capability to perform this trade-off analysis. This makes him more insistent on his requirements and generates a lobby for increased funding.

The above discussion does not imply that there is a lack of policy guidance mandating the need for trade-off analysis in the requirements process. Air Force Regulation 57-1 clearly states that "a major element of the requirements process is the continual identification of meaningful performance trade-offs whereby high-cost features providing only marginal performance gains are deleted from the system" [9]. However, implementation of this guidance in the earliest part of the requirements process is inconsistent. A Systems Management Acquisition inspection of requirements determination and validation observed a "...lack of a clearly defined, consistent, and well-integrated pre-SON process for identifying and rationalizing the requirement for a new or enhanced system and for accomplishing sufficient preliminary system and operational conceptualization to assess affordability, plan necessary technological improvements or adaptations, and support a program initiation decision" [10].

Both the literature and interviews with ESD personnel suggest that there is a tendency for the developer to be overly responsive to user requirements. One author stated that under the current process, "The user develops requirements, 'throws them over the wall' to the developer and the developer takes them as gospel" [11]. While this is obviously an exaggeration, it echoes the thinking that requirements should be "pure". It also highlights the need to take an integrated approach to the requirements process. The user, developer, and other commands involved in the requirements process must establish a working dialog early in the acquisition process. As mentioned above, each participant has a certain expertise necessary in the acquisition of a weapon system. The user knows what the system should do, the support commands understand suitability requirements, and the developer has the technical and financial expertise to help reconcile conflicting requirements given a constrained budget.

The developer can help all the participants in the requirement process realize that they are operating under a constrained budget. Resource boundaries can be estimated by using costs of similar programs to estimate current program costs. In addition, the developer can assess the political environment to determine where the particular program stands in relation to overall Department of Defense and Congressional priorities. Furthermore, the developer has the capability to perform trade-off analysis and this analysis should be used early in the requirements process to challenge user requirements. In this sense, design to cost means identifying key cost drivers and questioning whether high cost features are really necessary or just nice to have items. It also means assessing the level of a requirement. If the user specifies that the computational accuracy of a system must be to nine significant digits and current technology can only provide accuracy to five, then this requirement is subject to challenge on the basis that pushing the technology to meet the requirement will incur increased cost and risk.

The point of identifying cost drivers is not to force the user to accept current technology or reduce his requirement when the requirement is necessary to meet the threat. Only the user can provide a judgment on the utility of a requirement. The goal of this analysis is to inform the user of the cost implications of his requirement and allow him to re-evaluate his requirement in light of the new information. This is a subtle argument with which many users may take issue. However, the fact remains that users often employ conservative assumptions in their calculations. While this is prudent, it sometimes results in an aggregate outcome that is many times more conservative than each individual assumption [12].

Design to cost works best when the user clearly perceives a real resource constraint. Conditions that are favorable for using design to cost to prioritize requirements occur when the user needs a system in a relatively short period to meet an immediate threat. In this case, the user does not have time to lobby for increased funding and he must make do under the current resource constraint. Design to cost is also a helpful technique in discerning the user's priorities when it is apparent that political constraints will prevent funding beyond a certain level.

Identifying requirements that drive cost will not be a complete panacea in the effort to get users to prioritize all requirements, but it will reveal which high cost requirements the user thinks he must have in order to perform his mission. In addition, this type of design to cost is likely to aid in prioritizing operational performance requirements, but it will be less useful in prioritizing total system requirements.

In order to establish an overall view of the priority of both operational performance requirements and operational suitability requirements it is necessary to expand the design to cost concept from one of designing to production cost to one of designing to life cycle cost. Used during the requirements process, design to life cycle cost would identify cost drivers based on their contribution to both production cost and overall ownership cost (this includes operations and maintenance costs). This concept integrates all participants in the requirements process and helps all parties understand the relationship of their requirements to total system cost. Identification of key cost drivers early in the requirements process can help the user and support commands formulate alternative ways to meet a requirement at a lower cost.

2. Mandate Minimum Quantities

Design to cost and design to life cycle cost are techniques that the developer can employ to get a better understanding of the requirement priorities. However, these techniques may not be enough. As shown in figure 2, the user, developer, supporter, and trainer all interact in the requirements process. While the user generally has the upper hand in the relationship because he is the final judge of the military worth of a system, all of these participants have a horizontal relationship to each other with no participant in complete control.

Another way must be devised to make all the participants realize there is a resource constraint. Dr. Jacques S. Gansler, former Deputy Assistant Secretary of Defense for Material Acquisition, has suggested that one way to harmonize the conflicting interests is to get the participants to agree on an absolute minimum quantity of the system necessary to meet the threat [13]. Once quantity is established as a firm requirement, it becomes inviolate and other requirements will fall into place. Mandating a minimum quantity eventually relates to a rough fiscal constraint and trade-offs between other requirements and levels of requirements can be made in this context.

This approach makes sense, but minimum quantities are rarely explicitly agreed upon. Commitment to a minimum quantity is difficult for the developer because reduction in quantity is the program manager's ultimate fall back if the user will not make the appropriate trade-offs in requirements. In addition, if program uncertainties compound to cause unexpected costs flexibility in the quantity of a system procured may be the only solution.

3. Fix Authority for Determining Priorities

Even if all the participants recognize a valid resource constraint, this still may not be enough to elicit priorities. The Packard Commission assumes that all participants will react rationally to information on cost. However, the fact remains that different costs accrue to different participants. It may be rational for all the participants to insist on requirements that, by themselves, only marginally increase total cost. The result is that the sum of all the small increases in cost dramatically raises the price of the system. Moreover, it is not clear whose requirements should receive greater priority and reconciliation of these issues may only be possible by vesting authority to make decisions in a single body.

The Defense Science Board came to a similar conclusion in a 1986 study entitled "Practical Functional Performance Requirements". This study recognized the conflict between the participants responsible for operational suitability and the user's desire for performance. The Defense Science Board determined that the program manager (a member of the developer's organization) is best able to determine priorities. The board stated, "the need for specialized external review staffs can be reduced by assigning the program manager responsibility for the 'ilities' in the requirements document..." [14]. This conclusion makes sense because only the program manager has the capability to view requirements from all angles. The program manager must coordinate requirements with all the participants plus he and his staff have both the technical expertise and financial capability to understand how requirements interact.

C. <u>RECOMMENDATIONS</u>

The preceding section analyzed two general approaches to help prioritize requirements when many actors are involved in the requirement process. The first approach centered on making the participants realize the cost implications of their requirements and elicit priorities with this information. The second approach recognized that even with cost information, it still may be difficult to get all the participants to agree on a prioritized list of requirements. In this case, the use of a unitary decision maker was suggested.

<u>Recommendation 1:</u> Electronics System Division should concentrate on techniques that will help all the participants in the requirements process consider cost in determining requirement priorities.

ESD has the capability to use design to cost and design to life cycle cost techniques to elicit requirement priorities. While use of these techniques in the concept formulation phase may be limited to identifying cost drivers, this, in itself, goes a long way in determining the proper system trade-offs. ESD should also adopt a policy of making a minimum quantity an absolute requirement. In many instances, a minimum quantity equates to a rough performance requirement and other user requirements can then be rank ordered with this information.

Recommendation 2: ESD should not attempt to give program managers more authority to determine requirement priorities at this time.

Program managers are best situated to determine requirement priorities. However, ESD does not have the authority to institute this change. Participants in the requirements process may see this move as an attempt by the developer to usurp program authority. Furthermore, while recommending this change, the Defense Science Board concedes that it will require Secretary of Defense support [15]. Giving program managers more authority to prioritize requirements is a goal that must be pursued within a broader context of acquisition reform. For now program managers will have to rely on their persuasive skills to reach agreement on priorities. By employing design to cost and design to life cycle cost techniques, program managers will be in an improved bargaining position vis a vis other participants in the requirement process.

THE DEVELOPER AS SUPPLIER

A. ANALYSIS OF THE PROBLEM

Up to this point, the problems of C2 acquisition have been addressed on a traditional level. That is, most of the impediments to requirement prioritization stem from institutional arrangements, the requirements of the system are assumed to be known and relatively static, and the task for ESD is to devise a system to harmonize the various interests in the acquisition process.

This section deals with quite another problem that prevents requirement prioritization. The user does not know or cannot communicate his requirements. On the one hand, the user understands the operational environment, but he does not understand C2 technology well enough to employ its potential. On the other hand, the developer knows what C2 technology can do, but he does not understand the user's environment well enough to define and prioritize system requirements [16]. This problem usually occurs when a user is automating his command and control system for the first time, but it also happens when he is attempting a significant conversion of an old system and replacing it with state of the art technology. In both cases, requirement prioritization is hindered by three factors. First, the user does not understand the technology. Second, C2 technology is evolutionary in nature and requirements change as learning occurs. Third, a traditional acquisition approach retards the learning process.

1. Users Do Not Understand the Technology

The pace of computer and communications technology is moving so rapidly that it remains the exclusive domain of the technologist. Officers facing the daily pressure of command simply do not have the time to learn the technical aspects of the systems they need for command and control. Furthermore, many senior commanders spent their formative years using less sophisticated command and control systems to perform their mission. While these users understand the operational need for command and control systems, they simply do not have the capability to evaluate system level issues. This is a pressing concern because design issues such as distributed data base management and nodal versus fail-soft systems make a large difference how the system is employed and more importantly how it operates in war. An Armed Forces Communications and Electronics Association (AFCEA) study stated:

> A more severe "cultural" (or language) barrier exists between users and providers of C2 systems than exists between users and providers of ships, tanks, missiles, airplanes--for weapon systems, the user can relate to the meaning of a more maneuverable fighter plane, or a more accurate or longer range air-to-air missile, and can visualize the potential impact on mission performance more easily. Trying to understand, for example, what distributed microprocessor technology might mean to his ability to command and control is substantially more difficult, unless the user has had meaningful past experience with automated decision aids [17].

2. Uniqueness and Evolutionary Nature of C2 Technology

C2 systems are uniquely different from traditional weapons systems. C2 systems are "mind" extenders not "muscle" or "sense" extenders. C2 systems support the commander's decision process and as such the commander and his staff are an integral part of the system. Not only are C2 systems highly software intensive, but the software must interact with the cognitive process of the commander and his staff [18].

This interaction necessitates learning in the requirements process. As the user discovers the potential of new C2 technology, he may envision new ways to command his forces. This leads to changes in procedures and concepts of operation and then feeds back into the requirements process. As the user learns by operating the system, his requirements change. Therefore, a system built on pre-specified requirements may become rapidly obsolete if the user has not had the opportunity to react the technology and modify requirements before the system is developed [19]. Due to the nature and complexity of C2 technology, the user must gain hands on experience to define and prioritize his requirements [20]. Without this experience, learning will occur only when a final product is delivered. Refinements at this point are not only expensive, but disrupt system operation and degrade war fighting capability.

3. Problems Created by the Traditional Acquisition Structure

The traditional acquisition structure compounds problems of user learning by preventing the developer from establishing a true customer/supplier relationship with the user [21]. Under the traditional structure, the using commands have created planning organizations to study requirements and develop force employment concepts for new systems. These organizations are normally responsible for generating Statements of Need and system concept of operations. This arrangement has worked relatively well with systems that are well understood and incremental in nature. The concept of operations in these systems does not readily change and the planning organization can easily specify requirements based on past procedures. In aircraft acquisition for example, increased range, speed, and payload translates easily into requirements and the expected increase in capability fits in nicely with current force employment concepts. Requirements for these systems are relatively stable and predictable.

C2 systems do not enjoy stable and predictable requirements. In this case, the planning organization becomes a barrier between the developer and the real user [22]. While the planning organization acts as a user surrogate and frees the user to perform his daily tasks, it also insulates the real user from the very technology that he must understand. The planning organization becomes the main link to the developer. The result is that most of the technical expertise of the user is centered in the user surrogate. When the developer produces detailed design studies, it is the user surrogate who reads them. The real user generally has neither the time nor capability to understand the advice of the developer [23]. The user surrogate does attempt to keep the real user abreast of developments, but without a flesh and blood system to react to the real user is not inclined to change, or think about changing, his operating procedures. When the C2 system is finally delivered it, does not meet the real user's expectations. The end result is that the real user must adapt his procedures and concept of operations to the new system instead of evolving both the system and procedures together.

Another drawback of the user surrogate is also unique to C2 acquisition. C2 systems cross service boundaries and must be capable of interoperation with a multitude of other C2 systems. The user surrogate is typically service oriented and may not adequately consider the real user's interservice role when producing requirements [24]. Again, this problem is recognized and guidance explicitly directs all participants in the requirements process to address the interoperability issue [25]. However, the fact remains that the interoperability requirements are not adequately articulated. In many cases, only the real user understands the interoperability issue well enough to express it in the requirements process [26].

B. AN APPROACH TO THE PROBLEM

1. Use Prototypes to Define and Prioritize Requirements

Three things are apparent from the preceding section. First any attempt to define and prioritize C2 system requirements must start with the real user. Second, the real user must be educated so that he can communicate his requirements. Third, the developer and the real user (along with the user surrogate) must be wedded in a relationship that iterates requirements and promotes the transfer of information between all parties. Fortunately, prototyping is available to assist with all three of these problems.

Precisely because C2 systems involve a very real human element, operational prototypes, exercised in the user's environment are the best way to define and prioritize system requirements. Many C2 systems have been developed off-line to the user and have not integrated the personnel and procedural aspects of the real user's organization. This method of acquisition resulted in numerous C2 failures and will continue to threaten failure for future systems [27]. Operational prototypes can correct this problem. If operational prototypes are employed, the user will be able to "see, feel, touch, and taste" a proposed system. This, in itself, is an educational experience and can help the user adjust to new technology [28]. Paper studies and analyses do not promote this interaction. Use of prototypes can bridge the communication gap between developer and user. "With a prototype, the user can exercise the system just as though it were already operating in his own environment, and thereby provide vital feedback to the developer on the suitability of the specification" [29].

2. Costs and Benefits of Prototypes

Traditional C2 acquisition has shunned prototypes mainly because their cost is considered prohibitive [30]. Hassan Gommaa of the General Electric Company has concluded that prototypes can be developed for less than 10% of total software cost [31]. With software cost comprising 80% of total C2 system cost, this means prototypes can be developed for approximately 8% of total system cost [32]. This is a small price to pay to avoid developing a system with mis-specified or incomplete requirements. Errors in requirements are usually the last to be detected and the most costly to correct [33]. Adding non-monetary costs such as degradation of C2 systems vital to national security (detection and tracking of ballistic missiles comes to mind), strengthens the case for using prototypes to define and prioritize requirements.

Prototypes do not have to be full scale mock-ups of the system. The Armed Forces Communications and Electronics Association goes so far as to distinguish a rapid requirements definition capability as an entity distinct from a prototype [34]. Semantics aside, The AFCEA concept is what is implied by the word prototype in this report. Under this concept, off-the-shelf technology is rapidly assembled to simulate a number of functions the user wants to perform. The user employs the capability to solve real world problems and provides feedback to the developer. This iterative process refines the user's requirements to the point where they can be adequately specified to contractors in a request for proposal.

The benefits of using prototypes include: (1) involving the real user directly in the requirements process; (2) allowing the real user to adequately express his interoperability requirements; (3) identifying incorrect requirements; (4) identifying omissions in requirements; (5) identifying ambiguities in requirements; (6) eliminating misunderstandings between the developer and user due to different backgrounds; (7) providing insight on the proper system design; (8) facilitating user acceptance of new technology; (9) allowing the user to evolve his procedures along with the technology [35].

C. RECOMMENDATION

Recommendation 3 (see pages 23 and 24 for recommendations 1 and 2): Electronic Systems Division should develop a generic C2 prototyping capability which can be modified in response to initial requirements in the draft Statement of Need. This capability will allow prototypes to be developed and installed at the user's location to iterate requirements and determine priorities.

This approach can be successful only if ESD takes the initiative to train and educate the real user. ESD must also involve the surrogate user because many of the programmatic details must still be worked through him. It is envisioned that the use of prototypes will facilitate communication between the technologist, user, and surrogate user. The product of this communication will be well defined and prioritized system requirements. Furthermore, exercising the prototype in the user's environment should stimulate procedural thinking and help the user evolve his concept of operations in concert with advancing technology.

CONCLUSION

This paper makes a distinction between the developer's role as purchasing agent and the developer's role as supplier. The framework is useful because it allows the problems of requirement prioritization to be analyzed and it suggests solutions. In reality, the two roles of the developer are interwoven in the requirement process and the concept formulation phase. Their separation, while useful, does not tell the complete story.

Requirement prioritization must take place in an environment that integrates the perspectives of all the participants. Using the framework outlined in this paper may assist Electronic Systems Division to reach out to other participants in the requirement process, but too rigid an application could also separate the cost analyst from the technologist within ESD. In this light, cost and technical advice must remain coupled to help the user, user surrogate, and other participants in the requirement process determine their priorities.

ENDNOTES

1. Bengamin S. Blanchard, Design and Manage to Life Cycle Cost, (Portland: M/A Press, 1978), 14-15.

2. Alexander Kossiakoff, "Conception of New Defense Systems and the Role of Government R&D Centers," in The Genesis of New Weapons: Decision making for Military R&D, ed. Franklin A. Long and Judith Reppy, (New York: Pergamon Press, 1980), 77.

3. William L. Stanley and John L. Birkler, Improving Operational Suitability Through Better Requirements and Testing, (Santa Monica: RAND, 1986), 21.

4. Department of the Air Force, Air Force Regulation 57-1: Operational Needs, Requirements, and Concepts, (Washington D.C.: U.S. Government Printing Office, 1987), 2.

5. Department of Defense, Department of Defense Directive 5000.3, (Washington D.C.: U.S. Government Printing Office).

6. Michael Rich and Edmund Dews, Improving the Military Acquisition Process, (Santa Monica: RAND, 1986), 21-25.

7. Stanley and Birkler, 15.

8. President's Blue Ribbon Commission on Defense Management, A Formula for Action, (Washington, D.C.: U.S. Government Printing Office, 1986), 6.

9. Department of the Air Force, 2.

10. Air Force Inspection and Safety Center, Interim Report on System Acquisition Management Inspection of Requirements Determination and Validation, (Norton AFB, CA: Air Force Inspection and Safety Center, 1987), 3.

11. John C. Morgenstern, "C2 Systems Acquisition: The Requirements Problem," Signal, May 1983, 118.

12. Malcolm W. Hoag, Defense Economics in Action in America, (Santa Monica: RAND, 1968), 30.

13. Jacques S. Gansler, interview by author, 15 December 1987.

14. Defense Science Board, "Final Report of the 1985 Defense Science Board Summer Study on Practical Functional Performance Requirements," (Washington, D.C.: Office of the Under Secretary of Defense, Research & Engineering, 1985), 71.

15. Ibid., 3.

16. John C. Morgenstern, "C2 Systems Acquisition: The Requirements Problem," Signal, May 1983, 117.

17. Armed Forces Communications and Electronics Association, Command and Control (C2) Systems Acquisition Study: Final Report 1 September 1982, (Falls Church: AFCEA, 1982), I-12.

18. Ibid., I-11.

19. Ibid., V-13.

20. Ibid.

21. Robert B. Doane, "The Evolving Nature of the C3 Systems Acquisition Process," Concepts (Autumn 1982): 186.

22. Robert Kent, Interview by author, 9 February 1988.

23. Hassan Gomaa, and B.H. Scott. "Prototyping as a Tool in the Specification of User Requirements," In Proceedings of the 5th International Conference on Software Engineering, 9-12 March 1981, by the IEEE Computer Society (Los Alamitos, California: IEEE Computer Society, 1981), 333.

24. Program on Information Resources Policy, "Incidental Paper: Seminar on Command, Control, Communications and Intelligence," (Cambridge, Massachusetts: Harvard University Center for Information Policy Research, 1981), 102.

25. Department of the Air Force, 16.

26. Program on Information Resources Policy, 102.

27. Doane, 179.

28. Ibid., 183.

29. Gomaa, 334.

30. Ibid., 333.

31. Ibid.

32. Doane, 178.

33. Gomaa, 333.

34. Armed Forces Communications and Electronics Association, V-14.

35. Gomaa, 336-337.

SOURCES CONSULTED

Articles and Periodicals

Bruce, J. Gregor, and Dr. Alan M. Davis. "A New Look at the C3I software Life Cycle." Signal, April 1987, 85-93.

Brumm, Harold J. "Bureaucratic Competition and Weapon System Procurement." Defense Management Journal (Third Quarter 1986): 13-17.

Bush, Henry. "The C2 Communications Development and Acquisition Dilemma." Signal, January 1985, 25-33.

Doane, Robert B. "The Evolving Nature of the C3 Systems Acquisition Process." Concepts (Autumn 1982): 177-190.

Emmelhainz, Margaret A. "Innovative Contractual Approaches to Controlling Life Cycle Costs." Defense Management Journal (Second Quarter 1983): 36-42.

Hirsch, Edward. "Evolutionary Acquisition of Command and Control Systems." Signal, September 1985, 39-45.

Monteleon, Victor J. "Distributed C3 Test Bed and User Interactions." Signal, March 1987, 85-90.

Morgenstern, John C. "C2 Systems Acquisition: The Requirements Problem." Signal, May 1983, 117-122.

Orphanos, John A. "Management Fundamentals for Systems Acquisition." Signal, May 1986, 139-146.

Preston, Ray. "Baselining and Cost Caps: Putting the Lid on Weapon System Costs." Government Executive, February 1984, 28-34.

Sweet, Dr. Ricki and Dr. Armando LaForm Lopez. "Testing, the Modular C2 Evaluation Structure and the Acquisition Process." Signal, August 1987, 75-79.

Waks, Dr. Norman. "Inherent Conflicts in C2 systems Acquisition." Signal, May 1983, 83-93.

Wilson, Myron F. "Design to cost Application in Military Environment Means Changing Old Ways." Defense Management Journal (July 1973): 17-20.

Books

Blanchard, Benjamin S. Design and Manage to Life Cycle Cost. Portland, Oregon: M/A Press, 1978.

Cushman, John H. Command and Control of Theater Forces: Adequacy. Washington, D.C.: AFCEA International Press, 1985.

Hitch, Charles J. and Roland N. McKean. The Economics of Defense in the Nuclear Age. Cambridge, Massachusetts: Harvard University Press, 1960.

Jackson, Barbra Bund. Winning and Keeping Industrial Customers: The Dynamics of Customer Relationships. Lexington, Massachusetts: D.C. Heath & Company, 1985.

Leenders, Michiel R. and David L. Blenichorn. Reverse Marketing: the New Buyer-Supplier Relationship. New York: The Free Press, 1988.

Long, Franklin A. and Judith Reppy, eds. The Genesis of New Weapons: Decision Making for Military R & D. New York: Pergamon Press, 1980.

Margiotta, Franklin D. and Ralph Sanders, eds. Technology, Strategy and National Security. Washington, D.C.: National Defense University Press, 1985.

Seldon, Robert M. Life Cycle Costing: A Better Method of Government Procurement. Boulder, Colorado: Westview Press, 1979.

Interviews

Bethea, Colonel P., Tactical Air Command Liaison Officer, interview by author, 25 January 1988, Hanscom AFB, MA.

Doane, R.B., Senior Technical Advisor, Electronic Systems Division, Hanscom AFB, interview by author, 2 February 1988, Hanscom AFB, MA.

Field, J., research assistant to Professor Ronald Fox, Harvard Business School, interview by author, 26 October 1987, Cambridge, MA.

Gansler, J.S., Vice President and Director of the Analytic Sciences Corporation, telephone interview by author, 15 December 1987.

Goyette, Lieutenant Colonel R., Director of Airlift and Weather Systems, Electronic Systems Division, Hanscom AFB, interview by author, 24 November 1987, Hanscom AFB, MA.

Gudmundsson, B.I., Defense Procurement Analyst, Harvard University, interview by author, 5 January 1988, Cambridge, MA.

Kent, R., Program Manager for Cheyenne Mountain Complex, Electronic Systems Division, Hanscom AFB, interview by author, 9 February 1988, Hanscom AFB, MA.

LaBlonde, Colonel C.J., Assistant Deputy Commander for Development Plans and Support Systems, Electronic Systems Division, Hanscom AFB, interview by author, 3 November 1987, Hanscom AFB, MA.

Lafferty, Dr. E., software expert, The MITRE Corporation, interview by author, 9 February 1988, Bedford, MA.

Mleziva, M.L., Assistant Deputy Commander for Tactical Systems, J-TIDS, and AWACS, Electronic Systems Division, Hanscom AFB, interview by author, 8 December 1987, Hanscom AFB, MA.

O'Mahoney, T.P., Deputy Commander for Development Plans and Support Systems, Electronic Systems Division, Hanscom AFB, interview by author, 15 January 1988, Hanscom AFB, MA.

Salvucci, A.D., Assistant Deputy Commander for Strategic Systems, Electronic Systems Division, Hanscom AFB, interview by author, 14 December 1987, Hanscom AFB, MA.

Seifert, Colonel C., Base Commander of Hanscom AFB, interview by author, 11 December 1987, Hanscom AFB, MA.

Sherwood, R., Director of Planning Services, Electronic Systems Division, Hanscom AFB, interview by author, 3 November 1987, Hanscom AFB, MA.

Taylor, Lieutenant Colonel E., MILSTAR Program Office, Electronic Systems Division, Hanscom AFB, interview by author, 15 January 1988, Hanscom AFB, MA.

Volpe, D.J., Assistant Deputy Commander for International Programs, Electronic Systems Division, Hanscom AFB, interview by author, 17 December 1987, Hanscom AFB, MA.

Waks, Dr. N., Chief Management Scientist, The MITRE Corporation, interview by author, 23 February 1988, Bedford, MA.

Official Documents

Air Force Inspection and Safety Center. Interim Report on System Acquisition Management Inspection (SAMI) of Requirements Determination and Validation. Project Number 87-610. Norton AFB, California: AFISC, 23 February - 6 May 1987.

United States Department of the Air Force. Acquisition Program Baselining. AF Regulation 800-25. Washington, D.C.: Government Printing Office, 1986.

United States Department of the Air Force. Operational Needs, Requirements, and Concepts. AF Regulation 57-1. Washington, D.C.: Government Printing Office, 1987.

United States Department of the Air Force. System Operational Concepts. AF Regulation 55-24. Washington, D.C.: Government Printing Office, 1986.

United States Department of Defense. Department of Defense Directive 5000.3. Washington, D.C.: Government Printing Office.

Papers, Studies, and Proceedings

Armed Forces Communications and Electronics Association. "Command & Control Systems Acquisition Study: Final Report, 1 September 1982." Washington, D.C.: AFCEA International Press, 1982.

Defense Science Board. "Final Report of the 1985 Defense Science Board Summer Study on Practical Functional Performance Requirements". Washington, D.C.: Office of the Under Secretary of Defense, Research & Engineering, 1985.

Defense Science Board. "Report of the Defense Science task Force on Command and Control Systems Management". Washington, D.C.: Office of the Under Secretary of Defense, Research & Engineering, 1978.

Gomaa, Hassan and B.H. Scott. "Prototyping as a Tool in the Specification of User Requirements." In Proceedings of the 5th International Conference on Software Engineering, 9-12 March 1981, by the IEEE Computer Society. Los Alamitos, California: IEEE Computer Society, 1981, 333-342.

Hoag, Malcolm W. Defense Economics in Action in America. Santa Monica, California: RAND, 1968. P-3811.

Leong, Colonel Wah. The Role of the Operational Command in Acquiring C3 Systems. Maxwell AFB, Alabama: Air University Air War College, 1986.

Ottinger, Major Donald M. The Air Force Weapon System Requirements Process: Solutions to Recurring Problems. Maxwell AFB, Alabama: Air University Air Command and Staff College, 1986.

Pitterle, Captain Elise Killian. "An Identification of Operating and Support Cost Drivers for Command, Control, Communications, and Intelligence Systems." M.S. thesis, Air Force Institute of Technology, 1985. AD-A162-280.

President's Blue Ribbon Commission on Defense Management. A Formula for Action. Washington, D.C.: U.S. Government Printing Office, 1986.

Program on Information Resources Policy. "Incidental Paper: Seminar on Command, Control, Communications and Intelligence." Cambridge, Massachusetts: Harvard University Center for Information Policy Research, 1981.

Rich, Michael and Edmund Dews. Improving the Military Acquisition Process: Lessons from RAND Research. Santa Monica, California: RAND, 1986. R-3373-AF/RC.

Smith, G.K., A.A. Barbour, T.L. McNaugher, M.D. Rich, and W.L. Stanley. The Use of Prototypes in Weapons System Development. Santa Monica, California: RAND, 1981. R-2345-AF.

Stanley, William L. and John L. Birkler. Improving Operational Suitability Through Better Requirements and Testing. Santa Monica, California: RAND, 1986. R-3333-AF.

ATTACHMENT 7

DISTRIBUTION OF SONs, SORDs, and DSRDs

Draft SON, SORD and DSRD documents must be sent to commands and agencies listed below for review and comment. General Guidance includes: (1) The A is for Action addressee and are required responses. The I is for information and are not required responses. (2) Add HQ USAF/LEE and appropriate MAJCOM/DE when alteration of existing facilities, new facilities, land acquisition, relocatable buildings (including trailers), or temporary "facility substitutes" are involved; (3) Send a copy of all SONs/SORDs/DSRDs involving electronic combat system requirements or deficiencies to ASD/RW. (4) Operating commands should sanitize all copies according to AFR 80-11 before sending them to the Tri-Service Industry Information Offices.

NOTE: Changes and comments to this list should be directed to HQ USAF/RDQ.

TYPE OF ADDRESSEE (NUMBER OF COPIES)

	SON/SORD/DSRD COPIES	
ADDRESSEES	DRAFT	APPROVED/ VALIDATED
SAF/AQQX	A (1)	(1)
/AQRR	I (1)	(1)
/AQPM	I (1)	(1)
/AQSD	I (1)	(1)
/AQQT	I (1)	(3)
HQ USAF/XOEX	I (1)	(1)
/XOSX	I (1)	(1)
/XOXQ	I (25)	(25)
/XOXF	I (1)	(1)
/PRPRC	I (1)	(1)
/PRME	I (1)	(1)
/LEEX	I (1)	(1)
/LEXM	I (1)	(1)
/LEYY	A (5)	(5)
/LEYM	A (1)	(1)
/LE-RD	A (1)	(1)
/INEG	A (2)	(2)
/INT	I (1)	(1)
/INYX	I (3)	(3)
/IGF	I (1)	(1)
/DPXX	I (1)	(1)
/SAX	I (2)	(2)
/SCMR	I (3)	(3)
/REO	I (1)	(1)
Wash DC 20330		
HQ USAF/SGPT, Bolling AFB Wash DC 20332	I (1)	(2)
HQ USAF/FMC/Buzzards Point, Wash DC 20330	I (2)	(2)

TYPE OF ADDRESSEE (NUMBER OF COPIES)

ADDRESSEES	SON/SORD COPIES DRAFT	APPROVED/ VALIDATED
ANSER (Analytic Services Inc) Suite 800 1215 Jefferson Davis Highway, Arlington VA 22202	I (2)	(2)
HQ AFISC/SESD/IGSS Norton AFB CA 92409-7001	I (2 ea)	(2 ea)
HQ AFISC/SNA, Kirtland AFB NM 87117	A (5)	(5)
HQ AFSC/XRX, Andrews AFB DC 20334-5000	A (25)	(25)
HQ AFOTEC/XP, Kirtland AFB NM 87117	A (5)	(5)
FTD/XO, WPAFB OH 45433	I (4)	(4)
HQ ATC/TTY, Randolph AFB TX 78150-5001	A (6)	(6)
HQ AU-LSE-69-587, Maxwell AFB AL 36112	I (1)	(1)
HQ AFLC/XRI WPAFB OH 45433-5001	A (15)	(15)
OC-ALC/XRX, Tinker AFB OK 73145-5990	I (3)	(3)
SA-ALC/XRX, Kelly AFB TX 78241-5990	I (3)	(3)
SM-ALC/XRX, McClellan AFB CA 95652-5990	I (3)	(3)
OO-ALC/XRX, Hill AFB UT 84056-5990	I (3)	(3)
WR-ALC/XRX, Robins AFB GA 31098-5990	I (3)	(3)
AGMC/XRP, Newark AFS OH 43055-5105	I (3)	(3)
AFALC/LSX, WPAFB OH 45433-5000	I (10)	(10)
AFALC/LWE, Hanscom AFB MA 01731	I (1)	(1)
AD/AL Eglin AFB FL 32542	I (1)	(1)
HQ AMD/XR/RD Brooks AFB TX 78235-5000	I (1)	(1)
ASD/XR, WPAFB OH 45433	I (2)	(2)
BMO/MY, Norton AFB CA 92409-6468	I (1)	(1)
RADC/XPXS Griffiss AFB NY 13431-5700	I (5)	(5)
Joint Tactical C^3 Agency C^3A-ARM-M OASD-C^3I (ASC) Wash., DC 20301-3160	I (1)	(1)

(45)

TYPE OF ADDRESSEE (NUMBER OF COPIES)

ADDRESSEES	SON/SORD COPIES DRAFT	APPROVED/ VALIDATED
ANG/AFRES FWS, Tucson 1AP, Tucson AZ, 85734-1037	I (1)	(1)
DTIC/DDA, Cameron Station Alexandria VA 22304-6145	I (2)	(2)
ESD/XRX, ~~Hanscom AFB MA 01731-5000~~	I (5)	(5)
~~System Program Office (SPO) for Specific~~ System (TBD)	~~I (1)~~	~~(1)~~
Space Division/XR, PO Box 92960, Los Angeles CA 90009-2960	I (1)	(1)
HQ AFCC/XPQ, Scott AFB IL 62225-6001	A (6)	(6)
HQ AFIS/XPP, Ft Belvoir VA 22060-5788	I (4)	(4)
AWS/SYP, Scott AFB IL 62225-5000	I (2)	(2)
HQ ESC/XPX, San Antonio TX 78243-5000	I (6)	(6)
HQ AFRES/XP, Robins AFB GA 31098-6001	I (2)	(2)
NGB/RD, Wash DC 20310	I (3)	(3)
HQ AAC/PR, Elmendorf AFB AK 99506	I (3)	(3)
AFSPACECOM/XPX, Peterson AFB CO 80914	A (6)	(6)
HQ MAC/XPQ Scott AFB IL 62225-5001	A (6)	(6)
HQ PACAF/DOQ, Hickam AFB HI 96853-5001	A (9)	(9)
HQ SAC/XPRR Offutt AFB NE 68113	A (3)	(3)
HQ TAC/DRP - (SONs) /XPJ - (SORDs) Langley AFB, VA 23665-5001	A (3). A (6)	(3) (6)
HQ USAFE/DOQ, APO New York 09012-5001	A (10)	(10)
HQ AFOSP/SPD Kirtland AFB NM 87117-6001	I (2)	(2)

TYPE OF ADDRESSEE (NUMBER OF COPIES)

		SON/SORD COPIES
ADDRESSEES	DRAFT	VALIDATED
Tri-Service Industry Information Office, 5001 Eisenhower Ave, Alexandria VA 23333	I (3)	(3)
Tri-Service Industry Information Office, 1030 E. Green St, Pasadena, CA 91106	I (3)	(3)
Tri-Service Industry Information Office, AFWAL/MST Wright-Patterson AFB, OH 45433-6503	I (3)	(3)